In the Specification

On pages 8 and 9, please replace the paragraphs located between page 8, line 9 and page 10, line 3, with:

A variety of different terms are used in programming to describe different functional programming subunits. At different times and for different programming languages subunits of various sorts have been called functions, routines, subprograms, subroutines and other names. Such designations and the context or differences they represent are not significant to the present discussion and so this discussion is made simply in terms of programs, intending the term program to refer to functional programming units of any size that are sufficient to perform a defined task within a computer system or computing environment. Such specialized functions as those performed by macros within certain word processing programs, including, for example, in ~~Visual Basic~~ "VISUAL BASIC" macros for ~~Microsoft~~ Microsoft's "WORD" documents, are included within this general discussion. In this sense, individual documents may be considered to be programs within the context of this discussion.

For convenience and brevity, this discussion references viruses in the known sense of that term as being a self-propagating program generally undesired in the infected computer system. As used here, the term ~~Windows~~ "WINDOWS" is intended to reference any of the personal desktop operating systems sold by the Microsoft Corporation under the ~~Windows~~ "WINDOWS" brand name. The term PC or personal computer is used, unless specifically modified to indicate otherwise, to indicate a computer system based on the well-known x86 architecture, including those machines that presently are based on the microprocessor sold by Intel Corporation under its ~~Pentium~~ "PENTIUM" brand name and successors to that microprocessor and architecture. This discussion is provided to illustrate implementation of aspects of the invention. Aspects of the present invention find application in a range of different computer systems in addition to the illustrated personal computer systems.

The present inventor has analyzed the behavior of a variety of different viruses and other malignant source code. Certain general characteristics of viruses have been identified. A virus needs to infect other programs and eventually other programs to

propagate. Viruses consequently include infection loops that copy the virus into another executable program or sometimes into documents, in the exemplary case of ~~Visual Basic~~ "VISUAL BASIC" macro viruses. Viruses and trojans generally contain payloads. The payload allows the virus to affect the infected system or communicate its presence. A payload might be, for example, a message that pops up to announce the virus or a malicious function that damages the infected computer, for example by corrupting or erasing the data on the hard disk or by altering or disabling the BIOS within the BIOS flash or EEPROM.

On pages 17 and 18, please replace the paragraphs located on page 17, lines 1 and page 18, line 4 with:

~~Windows~~ "WINDOWS" 3.0 Operating System executables — NE-type executable which contains both the DOS MZ-header pointing at a DOS code area and a New Executable (NE) header containing the entry point of the ~~Windows~~ "WINDOWS" Operating System (protected mode) code. NE files are segmented.

"OS/2" Operating System executables — LE/LX type executable which contains both the DOS MZ-header and DOS code area and a protected mode section which is determined by the LE-header following the DOS code segment. Linear Executable (LE) files are used in the ~~Windows~~ "WINDOWS" 3 Operating System for system utilities and device drivers. LE files are segmented. LX files incorporate some differences in the way the page table is stored and are intended for the ~~OS/2~~ "OS/2" Operating System. LE files are

32-bit executables	segmented and the segments are paged. PE-type executable which contains both the DOS MZ-header and DOS code area and the Portable Executable header containing the entry point and file offset of the protected mode code. PE files are segmented.
OLE Compound Files	OLE compound files (COM) are document files that can contain executable format streams, usually referred to as Macros. All ~~Office~~ "OFFICE" program components incorporate ~~Visual Basic~~ "VISUAL BASIC" code for Applications, as does ~~Internet Explorer~~ "INTERNET EXPLORER" browser versions 4 and 5. ~~Windows 98~~ "WINDOWS98" operating systems can execute ~~Visual Basic~~ "VISUAL BASIC" code directly from a script file. The ~~Visual Basic~~ "VISUAL BASIC" code is compiled and stored in a stream, which is paged according to its file offset references stored in a linked list in the file header.

On pages 18 and 19, please replace the paragraph located between page 18, line 24 and page 19, line 11, with:

Compound document files can contain executable streams such as ~~Visual Basic~~ "VISUAL BASIC" code or macros. The structure of a compound document file is illustrated in the diagram shown in FIG. 3. The header of a compound document file contains a linked list (or File Allocation Table) which is referenced in a directory

structure that points to the entry point of the linked list. Each entry in the linked list refers to the next entry and a file offset. A value of -1 in the linked list indicates the end of a chain. Streams exist out of blocks, which may be scattered anywhere in the file in any order. In particularly preferred embodiments of the invention, code extracted from a compound document file is passed through a ~~Visual Basic~~ "VISUAL BASIC" decompiler before it is presented to a ~~Visual Basic~~ "VISUAL BASIC" emulator. Not all compound document files contain compiled ~~Visual Basic~~ "VISUAL BASIC" code. Hypertext markup language (HMTL) and ~~Visual Basic~~ "VISUAL BASIC" Script (VBS) files can contain ~~Visual Basic~~ "VISUAL BASIC" Script code as text. This code is preferably extracted and treated as a ~~Visual Basic~~ "VISUAL BASIC" Script code as text. This code is preferably extracted and treated as a ~~Visual Basic~~ "VISUAL BASIC" code stream within the virtual machine.

On page 27, please replace the paragraph located on page 27, lines 7-13 with

Macro viruses in documents are treated as if they were executables. The original ~~Visual Basic~~ "VISUAL BASIC" code is recovered by decryption (where applicable) and reverse compiling the ~~Visual Basic~~ "VISUAL BASIC" document (COM) stream. The resulting source code is neither saved nor shown to protect the rights of the original publishers of legitimate ~~Visual Basic~~ "VISUAL BASIC" software. After virtualization the source code is discarded.